Classification for diagnosis of Melanoma using Deep Learning

Abstract

Melanoma is one of the deadliest types of skin cancer affecting more than a million Americans every year. It is responsible for almost 90% of the skin cancer deaths. Early detection and treatment can help prevent the spread of melanoma throughout the body thereby increasing the survival rate of the patient. To cope up with the lack of adequate clinical expertise and the increasing cases of skin cancer, deep learning can play an important role in solving such critical issues. The objective of this project is to effectively distinguish between different skin lesions and correctly classify Melanoma with high accuracy. We build three such AI models and provide a comparative study between them to analyse the best performing model out of them.

1. Introduction

According to a study conducted by Skin Cancer Foundation [10], skin cancer is the most common cancer in the United States and worldwide as well. More than 2 people die of skin cancer in the US, every hour. Out of the different types of skin cancers, melanoma is the deadliest of all. Though the exact reason for its occurrence isn't clear, one of the predominant reasons could be exposure to ultraviolet radiation from sunlight which can increase the risk of developing melanoma. Melanomas mainly develop in areas that have had long exposure to the sun like the face, backs, arms, and legs.

Many studies have reiterated that Melanoma [6], though being a very deadly disease, is usually curable when detected and treated early and it becomes very difficult to treat once it has spread deeper into other parts of the body. When detected early, the 5-year survival rate for melanoma is 99 percent.

This poses a unique problem to solve. There may be multiple or different types of skin lesions. But it is crucial to identify the right type of lesion early and start the treatment accordingly. Usually, a clinical screening followed by a biopsy is performed to detect the disease. This cumbersome process can be made simpler by making use of AI models which can identify such skin lesions and detect them right away. We would like to solve this problem by building models that can detect Melanoma with the highest accuracies possible.

2. Background/Related Work

Various researchers have developed early detection techniques to identify Melanoma, a serious skin cancer type [1] [2]. In 1994, an Artificial Neural Network was trained on dermatoscopic images to differentiate Melanoma from Melanocytic Nevi and the results obtained were promising. However, like most earlier studies, this study lacked dermatoscopic images other than melanoma or nevi and had a small sample size. The research focused on melanocytic lesions (that is the differentiation between melanoma and nevi) and disregarded non-melanocytic pigmented lesions despite being common in practice [3].

The HAM10000 ("Human Against Machine with 10000 training images") dataset was introduced to boost the research on automated diagnosis of dermatoscopic images [3]. The HAM10000 dataset was collected from different populations and stored by different modalities. Our motivation to take up this problem statement was obtained from Kaggle Competition Skin Cancer MNIST: HAM10000 [3]. The goal of this challenge was to develop image analysis tools to enable the automated diagnosis of Melanoma from dermascopic images.

The results produced in [4] state that the best accuracies were obtained for SVM and Adaboost machine learning algorithms. The SVM model implemented in [5] was done using automated computer-aided-diagnosis on MATLAB and obtained very high accuracies when used as 4 class classifiers. However, the dataset used was very small. From [6] we know that SVM gives better accuracies for binary image classification. Deep convolutional neural networks [7] shows potential for highly variable tasks across many fine-grained object categories. The CNN achieves performance on par with all tested experts across both tasks, demonstrating an artificial intelligence capable of classifying skin cancer with a level of competence comparable to dermatologists. CNN implementation with dilated convolution, choose transfer learning VGG16, VGG19 and MobileNet gives good results [8] [9]. Deep models that network depth is of crucial importance [10] and based on this study we decide to

experiment with ResNet152 model as its complexity is simple compared to VGG16 and VGG19 [11].

3. Approach

For any image classification problem, once the data has been collected it needs to be pre-processed and cleaned for obtaining correct results. In our project, we have followed the steps as shown in the below flowchart.

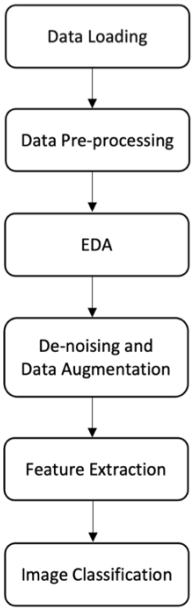

Fig 1. Classification pipeline

By making use of the input dataset, a deep learning model has to be built to accurately identify and classify the images corresponding to melanoma. The models that we have built include SVM classifier, CNN and ResNet-152. We expect Deep learning models – CNN and ResNet to give higher accuracy rates compared to the SVM classifiers for this image classification problem.

The classification accuracies of these models are then compared with each other, to evaluate the most suitable model for this problem. For further evaluation of the outcomes, the F1-scores are calculated. Intermediate evaluations also include graphs of loss vs epoch, train accuracy vs validation accuracy, confusion matrix and the classification report.

3.1 Data Loading

The dataset used for this project is the HAM- 10000 (Human Against Machine-10000) dataset from Harvard's Dataverse. This dataset consists of 10015 dermatoscopic images [7] from different populations, acquired and stored by different modalities. The different classes of images in this dataset include Actinic keratoses and intraepithelial carcinoma (akiec) (327), Basal cell carcinoma (bcc) (514), Benign keratosis-like lesions (bkl) (1099), Dermatofibroma (df) (115), Melanoma (mel) (1113), Melanocytic nevi (nv) (6705) and Vascular lesions (vasc) (142).

Metadata for these images (jpg files) consists of 5 features. They are dx (type of skin lesion; 7 classes), dx_type (technique used to identify lesion; 4 types), Age, Sex (male, female or unknown) and Localization (region where disease is present; 15 classes). The metadata for all these jpg images is present in the form a csv file. The jpg images and the metadata are loaded into the python notebook. Then, these images are mapped to their corresponding values from the metadata and stored in a data frame. This data frame will consist of the image dimensions (RGB channel) and its metadata which is to be used for further analysis.

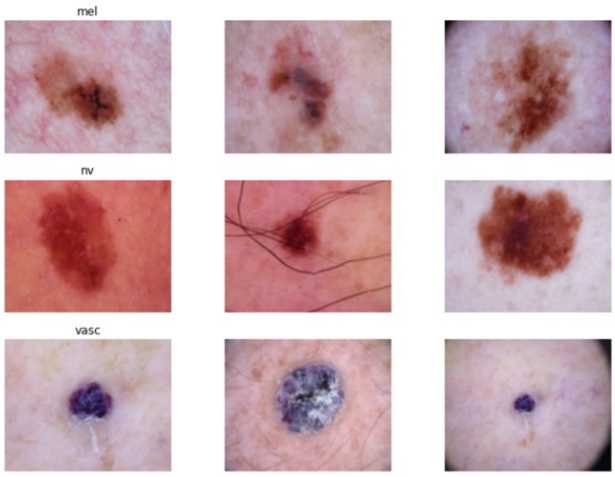

Fig 2. Part of loaded dataset from HAM-10000

3.2 Data Pre-processing

The loaded dataset has to be pre-processed so as to improve the quality of the images and the dataset to make it more suitable for feeding it into the model. This will help increase the accuracy and the efficiency of the machine learning/deep learning model.

As part of data pre-processing, the data was cleaned. 57 null values under the 'Age' feature were imputed by the mean value of the feature and unknown values under the 'Sex' feature were removed. Also, the images and metadata corresponding to Age = 0, were removed as it was arbitrary. Furthermore, for better clarity, the abbreviations of the 'Lesion Type' feature were expanded for all the images. The images were also resized from 450x600x3 to 64x64x3 so as to reduce the training time of the models.

3.3 Exploratory Data Analysis (EDA)

Exploratory data analysis can help detect obvious errors, identify outliers in datasets, understand

relationships, unearth important factors, find patterns within data, and provide new insights. The distribution of the cleaned data is visualized using EDA. Under this, univariate and bivariate analysis was used to analyse the features and the data distribution.

In Univariate analysis, frequency distribution of the data points based on age, gender, localization, disease type and method of detection were made. A few of the plots are shown as follows.

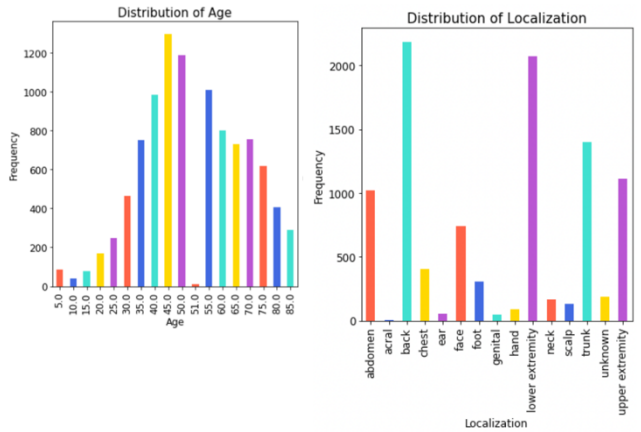

Fig 3. Univariate Analysis

From the univariate plots, the following were analysed:

- The highest number of skin lesions were found in the 45 years age group
- The occurrence of skin lesions were higher in males than females
- The skin lesions were predominantly seen in the back followed by the lower extremity regions
- The distribution of Melanocytic nevi (nv) class was the highest amongst all the other classes

In Bivariate analysis, plots of Localizations Vs Gender, Localizations Vs Skin Disease Type, Age Vs Skin Disease Type, Gender Vs Skin Disease Type. One such plot is as shown below.

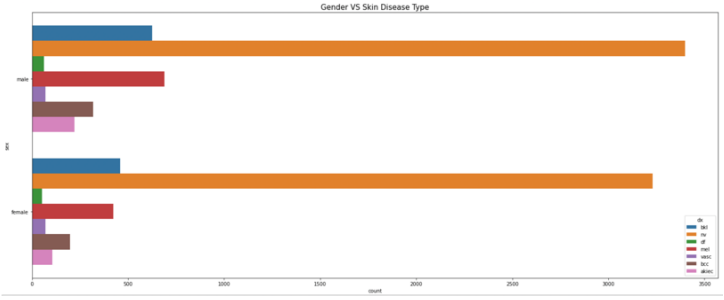

Fig 4. Bivariate Analysis – Gender vs Skin Disease Type

Notable preliminary analysis from the bivariate plots include:

- Skin diseases are more prominent in Men as compared to Women and other gender,
- Skin diseases are more visible on the "back" of the body and least on the "acral surfaces"(such as limbs, fingers, or ears) which happen to be more common amongst men, whereas the diseases are more common in the lower extremity of the body in women.

3.4 De-noising and Data Augmentation

On doing closer analysis of the images, it was seen that, the main lesions were occluded by hair. Removal of these hairs from the images can help increase the accuracy of classification. For hair removal, the image was first converted to grayscale, then blackhat filtering was performed on that image to find the hair contours. Then inpainting of the hair contours were performed by using the neighbouring pixels. This process of hair removal was all done using functions present in the OpenCV library. The output of the denoised images are as shown below.

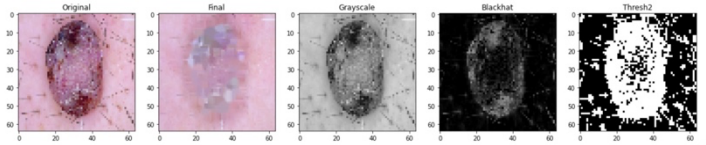

Fig 5. De-noising to remove hairs

For better classification results, it is best to have nearly equal amount of data samples for each of the class. In this dataset, it is seen that the samples for Melanocytic nevi (nv) is around 6000 whereas for all the other classes it is much lower (near 1000s and even 100s). Hence, data augmentation is performed for all the other classes apart from nv. The data augmentation techniques implemented include rotating by 90 degrees clockwise and anti-clockwise, rotating by 180 degree and flipping the image on the x-axis and y-axis. The total number of images have been increased from 10015 to 26376.

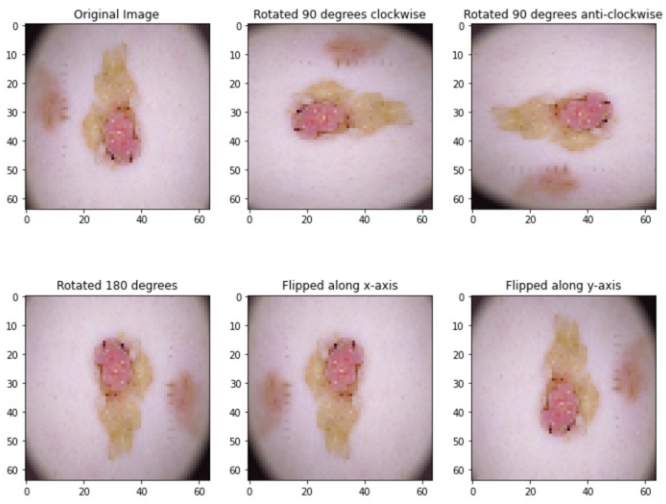

Fig 6. Data Augmentation

3.5 Feature Extraction

The features used for training the model include age, sex, localization and dx_type. The labels for these images are taken as dx. Feature importance plots were plotted using SHAP. These plots showed us the most important features which played a major role in the prediction of each class. For instance, it was observed that localization played a major role in the prediction of Melanoma (mel).

3.6 Image Classification

For image classification, the baseline model used is multiclass-SVM classifier. Two deep learning models using CNN classifier and ResNet-152 classifier are compared against the baseline model.

The labels for the different classes are 0 - Melanocytic nevi (nv), 1 - Melanoma (mel), 2 - Benign keratosis-like lesions (bkl), 3 - Basal cell carcinoma (bcc), 4 - Actinic keratoses and intraepithelial carcinoma (akiec), 5 - Vascular lesions (vasc) and 6 - Dermatofibroma (df).

3.6.1 Multiclass-SVM Classifier

Support Vector Machine (SVM) is a linear model used for classification and regression problems [5]. SVM classifies the data into multiple classes by creating hyperplanes.

Here, the dataset is split into train and test in the ratio 80:20 (train – 21252 and test – 5313). The training dataset is further split into train and validation (train – 20000 and validation – 1252). As part of pre-processing, the mean image was computed from the training dataset and was subtracted from the train, validation and test dataset. Before splitting the dataset, the data is also normalized to convert the original data to the range [0,1]. The loss function used here was the svm vectorized loss. The parameters – iterations, regularization strength and learning rate were fixed after performing hyperparameter tuning.

3.6.2 Convolutional Neural Network (CNN) Classifier

SVM performs better for binary image classification when compared to the multiclass image classification. Whereas CNNs shows better consistency in accuracy for both binary and multiclass image classification [12].

CNNs are made up of neurons having learnable weights and biases. It mainly consists of the following layers –

- Convolutional layer which is the main element of a network that performs the convolutional operations.
- Pooling layer down samples the spatial dimensions of the inputs given to it.
- Dense layer which connects the neurons between the different layers and usually forms the last few layers of the CNN model. Activation functions like ReLU are applied in this layer.
- Dropout layer is added to reduce overfitting of the model, by dropping a few neurons from the neural network during training.

For this project, the input size of the image is 64x64x3. Here, the dataset is split into train and test in the ratio 80:20 (train – 21252 and test – 5313). The training dataset is further split into train and validation (train – 20000 and validation – 1252).

The CNN architecture used in this project consists of the following layers –

1. Convolutional layer – filter size = 16, kernel size = 3, stride = 1 and padding = 1, activation = ‘relu’
2. MaxPool layer – pool size = 2
3. Convolutional layer – filter size = 32, kernel size = 3, stride = 1 and padding = 1 , activation = ‘relu’
4. MaxPool layer – pool size = 2
5. Convolutional layer – filter size = 64, kernel size = 3, stride = 1 and padding = 1, activation = ‘relu’
6. MaxPool layer – pool size = 2
7. Dense layer – hidden units = 512, activation = ‘relu’
8. Dropout layer – rate = 0.5
9. Dense layer – hidden units = 7, activation = ‘softmax’

The optimizer used for the CNN model is Adam with a learning rate of 1e-3. The loss was categorical cross entropy and accuracy was used as the metrics to evaluate the model. The CNN architecture was trained using the Keras and TensorFlow frameworks with 12.69 GB RAM. GPU was used on Colab Pro to train the model.

3.6.3 ResNet-152 Classifier

In CNN classifiers, beyond a particular point, the model faces the issue of vanishing gradient. This prevents the model from reaching a higher accuracy. Thus, ResNet-152 classifier model is used for this image classification problem. ResNet152 is considered over VGG19 as it has significantly increased depth but lower complexity [11].

We use a pre-trained ResNet-152 model from Keras. Architecture of ResNet-152: there are 9 convolutional layers in the first module, 24 convolutional layers in the second module, 108 convolutional layers in the third module and 9

convolutional layers in the fourth module. Combined with the first 7×7 convolutional layer and the last fully-connected layer, there are a total of 152 layers in this architecture.

For this project, the input size of the image is 64x64x3. Here, the dataset is split into train and test in the ratio 80:20 (train – 21252 and test – 5313). The training dataset is further split into train and validation (train – 20000 and validation – 1252). We added a few layers to the pre-trained model:

- Dense layer – hidden units = 4096, activation = 'relu'
- Dropout layer – rate = 0.4
- Dense layer – hidden units = 4096, activation = 'relu'
- Dropout layer – rate = 0.4
- Dense layer – hidden units = 7, activation = 'softmax'

The optimizer used for the ResNet-152 model is Adam with a learning rate of 1e-4. The loss was categorical cross entropy and accuracy was used as the metrics to evaluate the model.

4. Experiment

All the experimental results obtained as part of this project are discussed in this section. Section 4.1 describes the results of Multiclass-SVM classifier. Section 4.2 discusses the experimental analysis of the CNN classifier and Section 4.3 describes the experiments performed on ResNet-152 classifier. Finally, in Section 4.4, the comparisons between the three models are drawn.

4.1 Multiclass-SVM Classifier

After performing train, test and validation split, the Multiclass-SVM Classifier model was trained. The best validation accuracy was obtained for the hyperparameters - 2500 for iterations, 1e-7 learning rate and 1e+4 regularization strength. The training accuracy obtained was 52.33%, validation accuracy was 51.19% and test accuracy was 51.74%.

The loss value steeply decreased for the first 1000 iterations and became stagnant afterwards. Hence, the training was stopped after 2500 iterations. The Loss value Vs Iteration graph is seen in fig 7. The loss obtained after 2500 iterations was 3.229799.

The classification report for the test data is as shown in fig 8. Here, its seen that the f1-score is quite low for all the classes, with a score of 0.5 for melanoma. F1-score tells if the model correctly identifies real threats and is not disturbed by false alarms. This is extremely important for a skin cancer classification problem. A good F1-score is near 1 indicating perfect precision and recall and it will have near 0 value if precision and recall is zero.

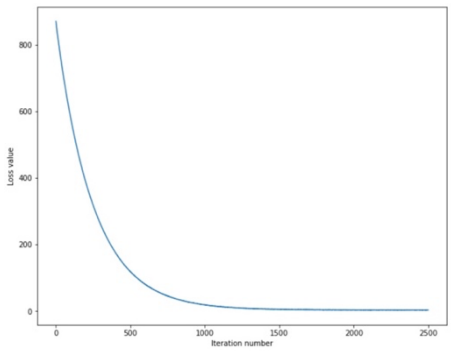

Fig 7. Loss Function

	precision	recall	f1-score	support
nv	0.32	0.29	0.31	428
mel	0.40	0.69	0.50	661
bkl	0.56	0.47	0.51	1284
bcc	0.03	0.01	0.01	136
akiec	0.62	0.64	0.63	1304
vasc	0.57	0.51	0.54	1324
df	0.40	0.29	0.34	176
accuracy			0.52	5313
macro avg	0.41	0.41	0.41	5313
weighted avg	0.52	0.52	0.51	5313

Fig 8. Classification report for SVM on test data

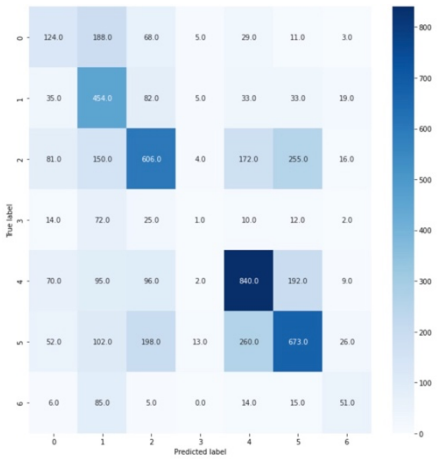

Fig 9. Confusion matrix for SVM on test data

From the confusion matrix, we can infer that the model almost predicts correctly for classes 1, 2, 4 and 5.

4.2 CNN Classifier

After performing train, test and validation split, the CNN Classifier was trained. We experimented with different number of layers and hyperparameters to check which architecture achieved the best results. We tried different values for dropout, different activation functions and added maxpool to reduce the computations. The best validation accuracy was obtained for the parameters – 64 batch size, 1e-3 learning rate and 50 epochs. The total trainable

parameters were 2,124,839. The architecture can be seen in the figure below.

```
Model: "model_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_2 (InputLayer)         [(None, 64, 64, 3)]       0

conv2d_3 (Conv2D)            (None, 64, 64, 16)        448

max_pooling2d_3 (MaxPooling  (None, 32, 32, 16)        0
2D)

conv2d_4 (Conv2D)            (None, 32, 32, 32)        4640

max_pooling2d_4 (MaxPooling  (None, 16, 16, 32)        0
2D)

conv2d_5 (Conv2D)            (None, 16, 16, 64)        18496

max_pooling2d_5 (MaxPooling  (None, 8, 8, 64)          0
2D)

flatten_1 (Flatten)          (None, 4096)              0

dense_2 (Dense)              (None, 512)               2097664

dropout_1 (Dropout)          (None, 512)               0

dense_3 (Dense)              (None, 7)                 3591

=================================================================
Total params: 2,124,839
Trainable params: 2,124,839
Non-trainable params: 0
_________________________________________________________________
```

Fig 10. CNN layers architecture

The training accuracy obtained was 70.17%, validation accuracy was 68.37% and test accuracy was 68.90%. The training loss obtained was 0.76 and the test loss obtained was 0.84.

From the following graphs (fig 11), we can see that, the validation accuracy is lesser than the training accuracy by a small margin, which shows that the model did not overfit. Furthermore, the loss is also decreasing with increase in epochs.

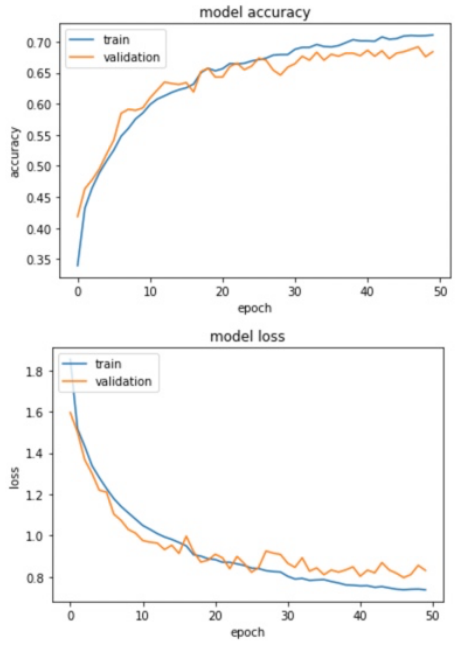

Fig 11. Accuracy Vs epoch and Loss Vs epoch

From fig 12, its seen that the f1-score and the accuracy has improved significantly when compared to the SVM baseline model for Melanoma class. From the confusion matrix in fig 13, we can infer that the model almost predicts correctly for classes 1, 2, 4 and 5.

	precision	recall	f1-score	support
nv	0.69	0.27	0.39	428
mel	0.87	0.54	0.66	661
bkl	0.56	0.82	0.66	1284
bcc	0.68	0.30	0.42	136
akiec	0.87	0.73	0.79	1304
vasc	0.65	0.75	0.69	1324
df	0.95	0.91	0.93	176
accuracy			0.69	5313
macro avg	0.75	0.62	0.65	5313
weighted avg	0.72	0.69	0.68	5313

Fig 12. Classification report for CNN on test data

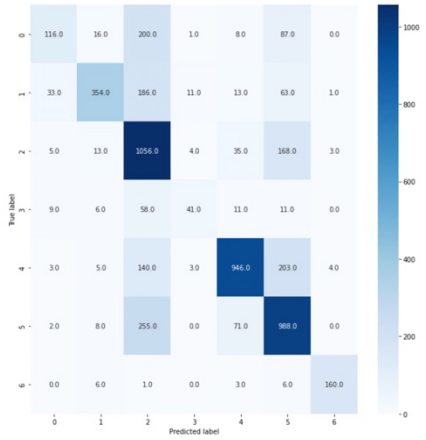

Fig 13. Confusion matrix for CNN on test data

4.3 ResNet-152 Classifier

After performing train, test and validation split, additional layers were added to the pretrained ResNet-152 model and it was trained. We experimented with different number of layers and hyperparameters to check which architecture achieved the best results. We tried different values for dropout and different activation functions to reduce the computations. The best validation accuracy was obtained for the parameters – 32 batch size, 1e-4 learning rate and 30 epochs.

The training accuracy obtained was 96.74%, validation accuracy was 91.82% and test accuracy was 92.02%. From the graphs (fig 14), we can see that, the validation accuracy is lesser than the training accuracy by a small margin, which shows that the model did not overfit. Furthermore, the loss is also decreasing with increase in epochs.

From fig 15, its seen that the f1-score and the accuracy has improved significantly when compared to the SVM baseline model and CNN for Melanoma class. All classes have near 1, F1-score. This shows that the model is highly accurate in classifying the correct classes. From the confusion matrix in fig 16, we can infer that the model almost predicts correctly for all classes from 0 to 6.

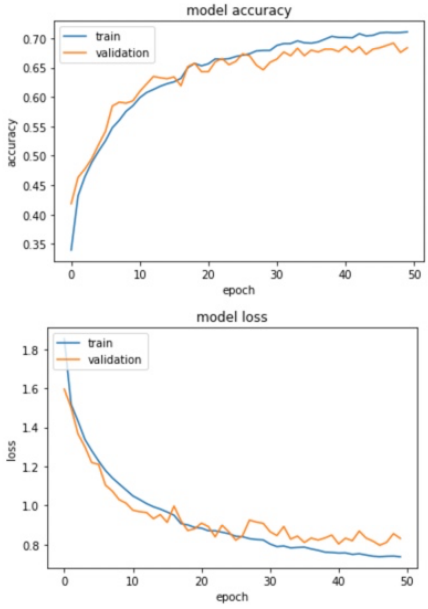

Fig 14. Accuracy Vs epoch and Loss Vs epoch

	precision	recall	f1-score	support
nv	0.85	0.99	0.92	428
mel	0.93	0.95	0.94	661
bkl	0.90	0.95	0.92	1284
bcc	0.96	0.99	0.98	136
akiec	0.96	0.83	0.89	1304
vasc	0.91	0.93	0.92	1324
df	0.98	0.99	0.98	176
accuracy			0.92	5313
macro avg	0.93	0.95	0.94	5313
weighted avg	0.92	0.92	0.92	5313

Fig 15. Classification report for CNN on test data

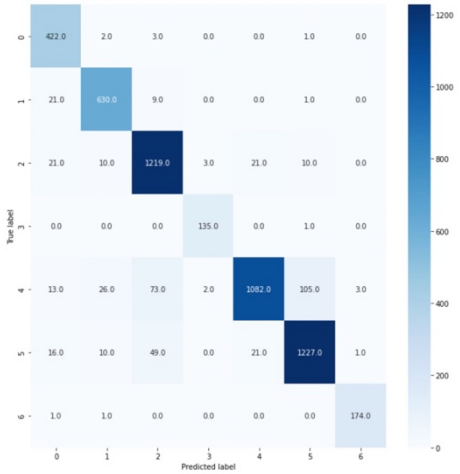

Fig 16. Confusion matrix for CNN on test data

4.4 Results

From the table 1, we observe that the ResNet-152 classifier model gives the best accuracy compared to the SVM baseline model and the CNN classifier model. ResNet-152 gives better accuracies as this deep learning model has increased depth compared to the other models as it has many layers. Despite the increased layers, ResNet-152 model has a lower complexity. We also observe that for all three models the test accuracy is only slightly lower than the train accuracy which indicates that the models do not overfit or underfit.

Model	SVM	CNN	ResNet-152
Training Accuracy	52.33%	70.17%	96.74%
Validation Accuracy	51.19%	68.37%	91.82%
Test Accuracy	51.74%	68.90%	92.02%

Table 1. Model accuracy comparisons

5. Conclusion

In this project, we have shown the performances of different machine learning and deep learning models specifically for this skin classification problem set. Even though SVM classifiers do perform well for binary classification, for multiclass image classification, it is always better to use complex deep learning image classification models like ResNet. We have also shown how deeper models (like ResNet-152) perform much better when compared to CNN models with few layers.

This problem set can be further expanded and developed into a working application to classify skin lesions in real time. Using tools like OpenCV, the trained models from this project can be used for performing live detection and classification of these skin lesions. This can act as additional tools to help clinicians in faster and early diagnosis of the disease and help treat skin cancer better.

6. References

[1] E. Jana, R. Subban and S. Saraswathi, "Research on Skin Cancer Cell Detection Using Image Processing," 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2017, pp. 1-8, doi: 10.1109/ICCIC.2017.8524554.

[2] Dildar M, Akram S, Irfan M, et al. Skin Cancer Detection: A Review Using Deep Learning Techniques. Int J Environ Res Public Health. 2021;18(10):5479. Published 2021 May 20. doi:10.3390/ijerph18105479

[3] Tschandl, Philipp, 2018, "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions", https://doi.org/10.7910/DVN/DBW86T, Harvard Dataverse, V3, UNF:6:/APKSsDGVDhwPBWzsStU5A==[fileUNF]

[4] A, Ameri. “A Deep Learning Approach to Skin Cancer Detection in Dermoscopy Images.” Journal of biomedical physics & engineering vol. 10,6 801-806. 1 Dec. 2020, doi:10.31661/jbpe.v0i0.2004-1107

[5] R. Suganya, "An automated computer aided diagnosis of skin lesions detection and classification for dermoscopy images," 2016 International Conference on Recent Trends in Information Technology (ICRTIT), 2016, pp. 1-5, doi: 10.1109/ICRTIT.2016.7569538.

[6] Jawale, Anupama & Magar, Ganesh. (2019). Comparison of Image Classification Techniques : Binary and Multiclass using Convolutional Neural Network and Support Vector Machines.

[7] Esteva, A., Kuprel, B., Novoa, R. et al. Dermatologist-level classification of skin cancer with deep neural networks. Nature 542, 115–118 (2017)

[8] Çevik, Emrah & Zengin, Kenan. (2019). Classification of Skin Lesions in Dermatoscopic Images with Deep Convolution Network. European Journal of Science and Technology. 309-318. 10.31590/ejosat.638247.

[9] Khan, Muhammad & Sharif, Muhammad & Akram, Tallha & Damasevicius, Robertas & Maskeliunas, Rytis. (2021). Skin Lesion Segmentation and Multiclass Classification Using Deep Learning Features and Improved Moth Flame Optimization. Diagnostics. 11. 811. 10.3390/diagnostics11050811.

[10] Md. Aminur Rab Ratul, M. Hamed Mozaffari, Won-Sook Lee, Enea Parimbelli (2020), Skin Lesions Classification Using Deep Learning Based on Dilated Convolution, doi: https://doi.org/10.1101/860700

[11] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[12] Brinker T, Hekler A, Utikal J, Grabe N, Schadendorf D, Klode J, Berking C, Steeb T, Enk A, von Kalle C, Skin Cancer Classification Using Convolutional Neural Networks: Systematic Review, J Med Internet Res 2018;20(10):e11936, URL: https://www.jmir.org/2018/10/e11936, DOI: 10.2196/11936